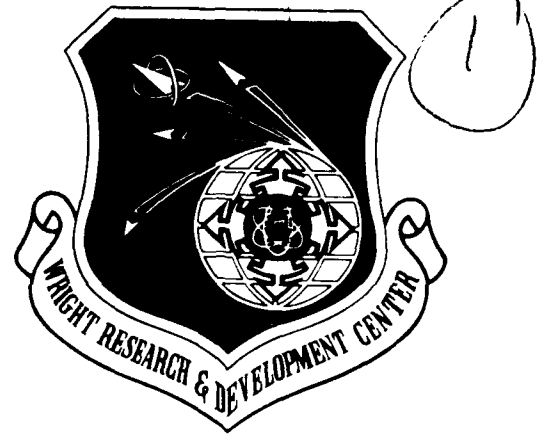


WRDC-TR-90-8007
Volume V
Part 37

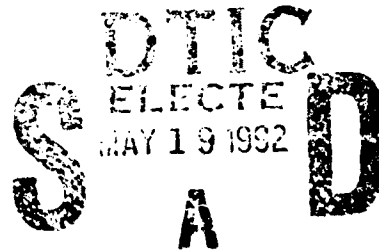
AD-A250 470



INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume V - Common Data Model Subsystem
Part 37 - DDL to NDDL Translator User's Manual

M. Apicella, S. Singh

Control Data Corporation
Integration Technology Services
2970 Presidential Drive
Fairborn, OH 45324-6209



September 1990

Final Report for Period 1 April 1987 - 31 December 1990

Approved for Public Release; Distribution is Unlimited

MANUFACTURING TECHNOLOGY DIRECTORATE
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6533

92-13191

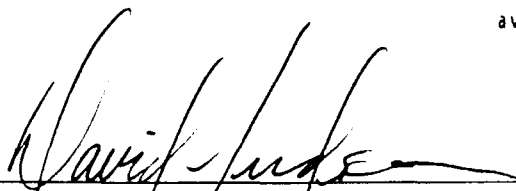


NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, regardless whether or not the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data. It should not, therefore, be construed or implied by any person, persons, or organization that the Government is licensing or conveying any rights or permission to manufacture, use, or market any patented invention that may in any way be related thereto.


This technical report has been reviewed and is approved for publication.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations


DAVID L. JUDSON, Project Manager
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

FOR THE COMMANDER:


BRUCE A. RASMUSSEN, Chief
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/MTI, Wright-Patterson Air Force Base, OH 45433-6533 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release; Distribution is Unlimited.		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE					
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UM 620341410			5. MONITORING ORGANIZATION REPORT NUMBER(S) WRDC-TR- 90-8007 Vol. V, Part 37		
6a. NAME OF PERFORMING ORGANIZATION Control Data Corporation; Integration Technology Services		6b. OFFICE SYMBOL (if applicable) WRDC/MTI		7a. NAME OF MONITORING ORGANIZATION WRDC/MTI	
6c. ADDRESS (City, State, and ZIP Code) 2970 Presidential Drive Fairborn, OH 45324-6209			7b. ADDRESS (City, State, and ZIP Code) WPAFB, OH 45433-6533		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Wright Research and Development Center, Air Force Systems Command, USAF		8b. OFFICE SYMBOL (if applicable) WRDC/MTI		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUM. F33600-87-C-0464	
8c. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB, Ohio 45433-6533			10. SOURCE OF FUNDING NOS.		
11. TITLE (Include Security Classification) See block 19			PROGRAM ELEMENT NO. 78011F	PROJECT NO. 595600	TASK NO. F95600
			WORK UNIT NO. 20950607		
12. PERSONAL AUTHOR(S) Control Data Corporation: Apicella, M. L., Singh, S.					
13a. TYPE OF REPORT Final Report		13b. TIME COVERED 4 / 1 / 87 - 12 / 31 / 90		14. DATE OF REPORT (Yr., Mo., Day) 1990 September 30	
15. PAGE COUNT 17					
16. SUPPLEMENTARY NOTES WRDC/MTI Project Priority 6203					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify block no.)		
FIELD	GROUP	SUB GR.			
1308	0905				
19. ABSTRACT (Continue on reverse if necessary and identify block number)					
<p>This document is to be used by the Common Data Model Administrator (CDMA) who is responsible for making changes to the CDM and ensuring it remains operable. The NDDL Translator consists of a translator for each DDL that must be translated to the Neutral Data Definition Language (NDDL).</p> <p>BLOCK 11:</p> <p>INTEGRATED INFORMATION SUPPORT SYSTEM</p> <p>Vol V - Common Data Model Subsystem</p> <p>Part 37 - DDL to NDDL Translator User's Manual</p>					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED x SAME AS RPT. DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL David L. Judson			22b. TELEPHONE NO. (Include Area Code) (513) 255-7371		22c. OFFICE SYMBOL WRDC/MTI

EDITION OF 1 JAN 73 IS OBSOLETE

Unclassified

DD FORM 1473, 83 APR

SECURITY CLASSIFICATION OF THIS PAGE

FOREWORD

This technical report covers work performed under Air Force Contract F33600-87-C-0464, DAPro Project. This contract is sponsored by the Manufacturing Technology Directorate, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Bruce A. Rasmussen, Branch Chief, Integration Technology Division, Manufacturing Technology Directorate, through Mr. David L. Judson, Project Manager. The Prime Contractor was Integration Technology Services, Software Programs Division, of the Control Data Corporation, Dayton, Ohio, under the direction of Mr. W. A. Osborne. The DAPro Project Manager for Control Data Corporation was Mr. Jimmy P. Maxwell.

The DAPro project was created to continue the development, test, and demonstration of the Integrated Information Support System (IISS). The IISS technology work comprises enhancements to IISS software and the establishment and operation of IISS test bed hardware and communications for developers and users.

The following list names the Control Data Corporation subcontractors and their contributing activities:

<u>SUBCONTRACTOR</u>	<u>ROLE</u>
Control Data Corporation	Responsible for the overall Common Data Model design development and implementation, IISS integration and test, and technology transfer of IISS.
D. Appleton Company	Responsible for providing software information services for the Common Data Model and IDEF1X integration methodology.
ONTEK	Responsible for defining and testing a representative integrated system base in Artificial Intelligence techniques to establish fitness for use.
Simpact Corporation	Responsible for Communication development.

Approved For	
NTIS - CRASH	J
DDO - JAG	J
Unpublished	J
Judson	
By	
DA-10-10-1	
Available	
Date	
A-1	

Structural Dynamics
Research Corporation

Responsible for User Interfaces,
Virtual Terminal Interface, and Network
Transaction Manager design,
development, implementation, and
support.

Arizona State University

Responsible for test bed operations
and support.

TABLE OF CONTENTS

	<u>Page</u>
SECTION 1.0 INTRODUCTION	1-1
SECTION 2.0 DOCUMENTS	2-1
2.1 Reference Documents	2-1
2.2 Terms and Abbreviations	2-1
SECTION 3.0 USE OF THE TRANSLATOR	3-1
3.1 Translator Input/Output	3-1
3.2 How To Run the NDDL Translator	3-1
3.3 Refinement of the Generated NDDL Code	3-2

APPENDICES

APPENDIX A	NDDL Grammar	A-1
B	DATABASE 2 Syntax	B-1
C	TOTAL DDL Syntax	C-1
D	User Interface Error Messages	D-1

FIGURES

<u>Figure</u>	<u>Title</u>	<u>Page</u>
3-1	Translator Input Screen	3-1

SECTION 1

INTRODUCTION

The NDDL Translator is a utility for translating the native Data Definition Language (DDL) for DATABASE 2 and TOTAL data base management systems into the IISS Neutral Data Definition Language (NDDL) for the creation of the Internal Schema (IS) specification for the Common Data Model (CDM).

The NDDL Translator consists of several translators, one for each DDL which must be translated to NDDL. Each translator differs only in its lexical analyzer and parser; so to avoid confusion this document will refer to these translators collectively as "the translator".

The translator accepts the subset of the native DDL which pertains to initial database definition. While all of the subset is accepted, only entities within the subset which have NDDL equivalents are translated (e.g. defining a file of a database is translated but the type of disk drive is accepted and ignored). The translator checks the syntax of the input but assumes the semantics are correct for the appropriate DDL. The grammar generated by the translator is given in Appendix A.

Intended Audience

This document is intended to be used by CDM administrators (CDMA), those who are responsible for making changes to the CDM and ensuring it remains in a consistent state.

SECTION 2
DOCUMENTS

2.1 Reference Documents

- [1] SYSTRAN, ICAM Documentation Standards, IDS150120000C, 15 September 1983.
- [2] D. Appleton Company, CDM Administrator's Manual, UM 620341000, 31 May 1988.
- [3] D. Appleton Company, CDM1, An IDEF1 Model of the Common Data Model, CCS 620341000, 31 May 1988.
- [4] Control Data Corporation, Neutral Data Definition Language User's Guide, 1 November, 1987.
- [5] C. J. Date, An Introduction to Database Systems, 1977, Addison-Wesley Publishing Company, Inc.
- [6] IBM, DATABASE 2 Reference Release 1.0, December 1984
- [7] Cincom Systems, TOTAL Database Administration Reference Manual, Release 8.1, 1978

2.2 Terms and Abbreviations

Application Process: (AP), a cohesive unit of software that can be initiated as a unit to perform some function or functions.

Common Data Model: (CDM), IISS subsystem that describes common data of an enterprise and includes conceptual, external and internal schemas and schema transformations.

Common Data Model Administrator: (CDMA), the person or group of persons responsible for creating and maintaining an enterprises's Common Data Model. The CDMA manages the common data rather than managing applications that access data.

Common Data Model Processor: (CDMP), a component of the Common Data Model subsystem which is the distributed database manager of the IISS.

Conceptual Schema: (CS), the standard definition used for all data in the CDM. It is based on IDEF1 information modelling.

External Schema: (ES), an application's view of the CDM's conceptual schema.

Integrated Information Support System: (IISS), a computing environment used to investigate, demonstrate, test the concepts and produce application for information management and information integration in the context of Aerospace

30 September 1990

Manufacturing. The IISS addresses the problems of integration of data resident on heterogeneous data bases supported by heterogeneous computers interconnected via a Local Area Network.

Internal Schema: (IS), the definition of the internal model, the storage structure definition, which specifies how the physical data are stored and how they can be accessed. It is represented in terms of the physical database components, including record types and inter-record relationships.

Neutral Data Definition Language: (NDDL), A language used to manipulate and populate information in the Common Data Model (CDM) or IISS System Database.

Neutral Data Manipulation Language: (NDML), A language developed by the IISS project to provide uniform access to common data, regardless of database manager or distribution criteria. It provides distributed retrieval, single node update, and non-guaranteed distributed update.

Presentation Schema: (PS), The totality of the form fields in an application which are targets of data derivative from the common data.

SECTION 3

USE OF THE TRANSLATOR

3.1 Translator Input/Output

The translator takes as input the DDL of either DATABASE 2 or TOTAL. The input is restricted to functions which pertain to initial database definition and not incremental changes. The grammar of each Data Base Management System (DBMS) the translator accepts is listed in Appendices B and C.

The output of the translator is a source text file of NDDL which pertains to the definition of an IS for the CDM. The output source will be syntactically correct but may be semantically incomplete. Missing items are flagged in the generated source by comments and in some cases will be supplied with dummy default values. The CDMA must supply such things as the NTM host. The forms interface to the NDDL processor (another CPCI) allows you to perform stepwise refinement of the output NDDL in order to semantically complete the NDDL. The grammar for the output of the translator is given in Appendix A.

3.2 How to Run the NDDL Translator Program

The translator is run from the IISS Function Screen. Its name is "DB2TRANZ" for DATABASE 2 and "TOTTRANZ" for TOTAL. When the translator starts, the form shown in Figure 3-1 is presented. The CDMA enters the name of the input file to be translated and the name of the output file to receive the translated source.

DDL TO NDDL TRANSLATOR	
Input File	_____
Output File	_____
Msg: _____ application	

Figure 3-1 User Interface Screen for Translator

Error messages from the translator are presented in the message line and are of the following form:

error line number: message text

where the error type is either WARNING, ERROR, or FATAL. A WARNING message indicates a problem but does not prevent translation to NDDL. An ERROR message indicates a problem which does prevent translation. After an error is found the translator continues reading the input file so that further errors in the text (if there are any) can be found; but no NDDL translation is generated. A FATAL message indicates a problem which prevents translation and causes the termination of the translator. The line number is the number of the line in the input file where the error occurred. The message text is a description of the error. A list of error messages is documented in Appendix D.

3.3 Refinement of the Generated NDDL Code

Since the generated NDDL code is sometimes semantically incomplete, the NDDL code is refined via the NDDL interactive processor with translator-issued modification requirements as a guide. The modification requirements are in the form of comments embedded in the generated NDDL. They immediately follow the entry to be modified. Their format is as follows:

/* MOD - message text */

These messages are documented along with the related NDDL statements in Appendix A. Some examples are:

For DATABASE 2:

```
define DB2 database named DSN8DPRG on host dummy;  
/* MOD - host name */
```

For TOTAL:

```
define TOTAL database named ORDRDB on host dummy;  
/* MOD - host name */
```

For more information concerning the NDDL interactive processor refer to the NDDL User's Guide for that module.

APPENDIX A

NDDL Grammar

These are the NDDL commands that may be generated by the translator. Each command is listed for DATABASE 2 and TOTAL. Version 2.3 of NDDL is used. In the following a word that begins with a capital letter is a keyword and must be typed in exactly as it appears. An all lower case word is a name supplied by the user. A token which appears in braces { } indicates one must be selected. A token which appears in brackets [] indicates the entity is optional. A token which is followed by an elipsis indicates the entity may be repeated.

Define Database

DB2

- Define DB2 Database Named db_name On Host host_name;

Note: A modification requirement will be issued for the user to supply a host name.

TOTAL

- Define TOTAL Database Named db_name On Host host_name
With Files file_name ... ;

Note: A modification requirement will be issued for the user to supply a host name.

Define Table

DB2

- Define Table table_id
With Columns
(column_name Datatype data_type_name) ... ;

Note: SQL data types will be translated as follows:

<u>SQL</u>	<u>NDDL</u>
INTEGER	INTEGER
SMALLINT	SMALLINT
FLOAT	FLOAT
DECIMAL(n,m)	DECIMALnnn_mmm
CHAR(n)	CHARnnn
VARCHAR(n)	VARCHARnnn
LONGVARCHAR	LONGVARCHAR

TOTAL

```
- Define Table table_id
  With Columns
    / [level_no] column_name_1
    / Datatype data_type_name [Unique Key] \
    < Redefines column_name_2 >
    \ Unknown
    /
    [level_no] Filler filler_size
    \
  ... ;
```

Note: A modification requirement will be issued for each filler field to use a named field.

Define Set

DB2

- Not Applicable

TOTAL

```
- Define Set set_name
  From table_id1 to table_id2 ...
  Linked By column_name;
```

Note: column_name is from table_id1.

Describe

```
- Describe Comment_On Of
  / Table Class table_id \
  \ Column Class column_name /
  "string" ;
```

APPENDIX B

DATABASE 2 Syntax

The following are DATABASE 2 commands that could appear in a database definition. In the following a word that begins with a capital letter is a keyword and must be typed in exactly as it appears. An all lower case word is a name supplied by the user. A token which appears in braces { } indicates one must be selected. A token which appears in brackets [] indicates the entity is optional. An entity which is followed by an elipsis indicates the token may be repeated. Note that several databases and tables may be defined in one source file.

```
Create Database (maps to NDDL Define Database)
  Create Database db_name
  ...
```

```
Create Table (maps to NDDL Define Table)
  Create Table table_id
    ( { column_name data_type [ Not Null ] } ,... )
    / In [ db_name. ]_tablespace_name \
    \ In Database db_name /
  ...
```

```
Comment On (maps to NDDL Describe)
  Comment On / Table table_id \
    \ Column table_id.column_name /
  "string"
```

Commands that may appear in DDL.

```
Create Index
Create Stogroup
Create Synonym
Create Tablespace
Create View
Grant (privileges)
```

APPENDIX C

Total DDL Syntax

The following are statements from the TOTAL DDL which could appear in a database definition. TOTAL is a fixed format line oriented language and could be parsed by little more than a lexical analyzer. Statements which map to NDDL are prefixed by a star. In the following a word that begins with a capital letter is a keyword and must be typed in exactly as it appears. An all lower case word is a name supplied by the user. A name of mmmm indicates a master table name. A name of vvvv indicates a variable table name. A token which appears in braces { } indicates one must be selected. A token which appears in brackets [] indicates the entity is optional. A token which is followed by an elipsis indicates the entity may be repeated. The notation ".pp." indicates a level number for a column (e.g. a COBOL record).

Note that one database and several master and variable tables may be defined in one source file. All TOTAL data fields will be of the data type CHAR n (where n is the size in bytes of the field. A warning will be issued for the user to modify these fields. All filler fields will be flagged as warnings as fillers should not be used. The data definition should not contain fillers, use the database definition which has all fields named. In variable records the record code redefines the last data field in the base data section. The level numbers will be incremented by one to handle redefinition of variable data areas. Link and key fields which have secondary names will have the secondary name treated as a redefinition of the field.

```
BEGIN-DATA-BASE-GENERATION:
OPTIONS: ...
LOGGING: ...
CTLX: ...
* DATA-BASE-NAME = xxxxxxxx ; Maps to NDDL Define Database
SHARE-IO: ...
IOAREA = ...
JCL = ...

BEGIN-MASTER-DATA-SET:
* DATA-SET-NAME = mmmm ; Maps to Define Database
IOAREA = xxxx

; Maps to Define Table

MASTER-DATA:
* [.pp.] mmmmROOT = 8 ; named data field
* [.pp.] mmmmCTRL = n ; named data, key field
* [[.pp.] mmmmLKxx = 8] ; named data, link field
* [[.pp.] xxxxxxxx [= n]] ; named data or unknown field
* [[.pp.] *FILLER* = n] ; unnamed data field
END-DATA:

; Physical data description
DEVICE = ...
ACCESS-METHOD = ...
TOTAL- LOGICAL-RECORD = ...
TOTAL-TRACKS = ...
```



```
LOGICAL-RECORD-LENGTH = ...
LOGICAL-BLOCKS-PER-TRACK = ...
CONTROL-INTERVAL = ...
CONTROL-INTERVAL-SIZE = ...
DISK-EXTENTS = ...
DOS = ...
OLD-FILE = ...
END-MASTER-DATA-SET:

BEGIN-VARIABLE-ENTRY-DATA-SET:
* DATA-SET-NAME = vvvv          ; File of Define Database
  IOAREA = ...

                                ; Maps to Define Table

BASE-DATA:
* [[.pp.] vvvvCODE = 2]          ; named data field
* [[.pp.] xxxxxxxx = n [=mmmCTRL] ; named data field associated
                                ; with key
* [[.pp.] *FILLER* = n]          ; unnamed data field
* [[.pp.] mmmLkxx = 8 [=xxxxxxx] ; named data, link field of
                                ; master record
* RECORD-CODE = xx              ; named redefines field
* [[.pp.] xxxxxxxx = n [=mmmCTRL] ; named data field associated
                                ; with key ? is it unique?
* [[.pp.] *FILLER* = n]          ; unnamed data field
* [[.pp.] mmmLkxx = 8 [=xxxxxxx] ; link field of master record

END-DATA:
                                ; Physical data description

DEVICE = ...
ACCESS-METHOD = ...
TOTAL- LOGICAL-RECORD = ...
TOTAL-TRACKS = ...
LOGICAL-RECORD-LENGTH = ...
LOGICAL-BLOCKS-PER-TRACK = ...
CONTROL-INTERVAL = ...
CONTROL-INTERVAL-SIZE = ...
DISK-EXTENTS = ...
DOS = ...
OLD-FILE = ...
END-VARIABLE-ENTRY-DATA-SET:

END-DATA-BASE-GENERATION:
```

APPENDIX D

User Interface Error Messages

The following are a list of error messages which could be presented when using the translator. A percent sign (%) preceding a name indicates the value of the name will be substituted.

WARNING n: string too long
A string is longer than 160 characters.

ERROR n: Unable to open %file_name
The translator was unable to open the file.

ERROR n: unterminated string
A string has no closing quote.

ERROR n: syntax error
There is a syntax error on the given line.

FATAL n: out of memory
The translator was unable to allocate sufficient memory.